An Interview with

FERNANDO J. CORBATO

OH 162

Conducted by Arthur L. Norberg

on

18 April 1989
14 November 1990

Cambridge, MA

Charles Babbage Institute
Center for the History of Information Processing
University of Minnesota, Minneapolis
Copyright, Charles Babbage Institute

NORBERG: Who was that?

CORBATO: Frank Verzuh. I in turn replaced him. I do not know if my title changed or not; I forget. It might have been assistant or associate director, but I took on the role of basically being Morse's acting director. Morse's style of management was he would put in one day a week at it. He would show up one day a week, and of course, if there was a crisis or a problem he was on the phone. So that meant the person who was running it for him had a lot of leeway to do things, and a lot of responsibility to make things happen, because we were also running the Computing Center with a fairly hig staff, operators and programmers...

NORBERG: I guess I am a little confused. I did not realize that there was a difference hetween the Computation Center and the Computing Center.

CORBATO: I am sorry. It was the same. I use the word loosely. Formerly it was the Computation Center. In the early days there was only one such place. For many years it had a de facto monopoly, which gradually eroded away with the advent of, basically, mini-computers.

NORBERG: So this monopoly would have lasted about ten years then, if that was the case? Don't you think?

CORBATO: It did not last quite that long, but that is approximately right. People were making special case arguments about why they had to have their own equipment.

NORBERG: Who was the first one to get another set of equipment? Do you remember?

CORBATO: I am not sure I have got the first one nailed down. One of the consequences of working with IBM was that the center would keep upgrading its equipment. IBM retained title on the equipment, and we would jawbone them until they would finally relent and give us a slightly better machine. It was never quite as early as

we wanted it. So we went from a 704 to a 709, and then later we went to a 7090. The 709 that was released was bought hy John Slater's group. They claimed that they could somehow run it economically themselves, giving themselves wholesale rates. Michael Barnett was the person who promised to run it. The trouhle, of course, was that it was a tuhe machine, hasically past its useful life. And even though MIT didn't pay a whopping price, they chose to charge themselves two million. They never paid it off, I understood. So it was a had husiness deal.

NORBERG: Besides allocating research assistantships and managing the organization, what sorts of projects did you get involved in then that would have led to the CTSS project later on? Let me ask the question differently. It will prohahly he more intelligent, too. As machines like the 704 were hrought into the campus and replaced WHIRLWIND 1, WHIRLWIND 2...

CORBATO: No, there was never a two.

NORBERG: Oh, okay, and replaced WHIRLWIND 1, how did you people organize the center to improve its effectiveness, increase the reliahility of the machinery, increase use effectiveness in productivity, and so on?

CORBATO: Well, we tried to run a state-of-the-art center. For instance, I used to go to the so-called Share meetings. It was an organization of IBM big machine scientific users. It probahly happened twice a year. You would compare notes with peers, hoth university and industrial. So one had an idea of what it meant to run a state-of-the-art center. In particular, what happened rapidly with that class of machine was that they evolved into a batch processing mode of operation. The initial configuration of the machine was to use a card reader and an on-line printer, on-line punch. The trouhle was that the machine was paced hy the speed of those. Since there was no other way of getting information in or out -- one normally needed a lot of information -- the result was that the machine worked at a crawl unless you did something ahout that. So the solution that came from batch processing was that all the input decks would he prerecorded on magnetic tape using an auxiliary machine, which in turn would serve as the input to the 704. It would run through a sequence of johs and regardless of what happened, whether it would

crasb or wbetber they ran to completion, the results would be pumped out to another magnetic tape, again for subsequent off-line printing and punching on it either on another or on that same auxiliary machine. The only trouhle with that was that by hatching things up it meant that the cycle time for putting in an input and receiving a result was, at a minimum, a balf an bour; more often it took a couple bours. Sometimes it even took a day, because being a single queue, the delay time in getting jobs back would slide to a day. That gradually happened more frequently, although we went to great effort to try to work our way out of it hy having priority queues, short jobs, and deferring long johs to the weekend. At some point, the load huilt up to tbe point wbere even the sbort johs were getting terrible service. At the same time people were heginning to write more and more intricate programs. As people got more amhitious writing programs, they ohviously had more and more trouhle not making a mistake somewhere, and the dehugging process got to be more and more frustrating. So this was reaching a bead. In some ways the universities were tbe first to notice the problem, because they were the least wealthy, and I recall that places like United Aircraft down in Hartford, whicb were doing a lot of military defense work, were using five 709s at one time, and we had one shift of one (laugh). So it was pretty desperate. Well, where do you want me to go next? Sbould I discuss the kind of projects we were working on?

NORBERG: Yes. Well, wben I asked you the question about trying to increase tbe productivity of the machinery and so on, you defined the problem all rigbt -- that tbe problem just kept getting worse and worse because of both the intricacies of the programs that were heing run, and also the increase in the number of users. Now, you were faced with this situation in ahout 1958-59, and you could not buy anotber machine.

CORBATO: That is right. We could not convince IBM to do the upkeep and give us more resources. They tried to be generous in their own way, but tbey had their own internal fights, I am sure.

NORBERG: Was there any attempt to find resources elsewhere?

CORBATO: The quick answer is no. We were pretty much locked into working with IBM in that center, because we were given the use of the machine, and we in turn gave... We allowed it to be used free of charge, so one of the auxiliary complications was that we could not buy our way out. The users were used to free computing time, and they did not have funds in their research budgets to pay for computing. So if we were to try to get a new machine, even if it was another IBM machine, and suddenly started charging for it, people would not have the money to pay for it. That was a situation that did not get rectified until about 1965. It was one of the things that additionally aggravated the problem. The first genesis of timesharing began when John McCarthy wrote a key memo, which suggested that one could set up a dehugging terminal that could interrupt the main computer without any damage to the job in progress, do some computations, and interact with the terminal and at the same time allow the person to run a program to check out debugging problems. John has told me that he went to a UNESCO conference and heard Christopher Strachey say something very similar; however when I looked at the documents that had gotten written up, it is clear that John had a broader vision. He really recognized that you could have lots of terminals. He had a whole framework, a whole computational environment. Strachey, I think, thought of it mostly as a debugging terminal. So again, those are written documents.

NORBERG: Yes, I have read the Strachey piece. I have the McCarthy piece, for that matter.

CORBATO: Yes, now they were a vision.

TAPE 1/SIDE 2

CORBATO: In the early 1950s, there were many universities that had been involved in building computers, but hy the mid-1950s, as the commercial world began to build up, the university activities were dying down. The companies, however, took a kind of closed-box view of computing equipment -- they didn't want you messing around with their circuits, or they refused to repair it anymore. So, in fact, to get any changes made to the IBM equipment we had to go through a long, circuitous route submitting, in the jargon of IBM, a "request for price

quotation" -- RPQ -- which meant that you wrote down your engineering change idea that you would like. They in turn would send it back to engineering, and depending on whether they wanted to or not, would quote you a price which was either outrageous or modest. It was a combination of IBM arguing within itself between engineering wanting to do it and marketing pushing it or not. So that was the path we had to argue for to get any changes made to the 704. We began to do that, and the local support groups within IBM were quite helpful, but the engineering people were not. They were more preoccupied with trying to build machines that went out to the marketplace. But fortunately, enough of the other aerospace type companies had asked for special changes that we were, by and large, able to find previously done changes which we could ask to bave done for us too.

NORBERG: I see. Can I interrupt with two questions then? One of them is, bow early did tbese requests for price quotations begin to be sent to IBM?

CORBATO: Well, somewhere around 1958-59. After the McCarthy memo. Well, we might have asked for one or two on that special...

NORBERG: Yes, that was going to be my second question.

CORBATO: I think we had a special one or two right from the beginning, but the ones focused on timesharing did not begin until after the McCarthy memo. The key ingredients that we foresaw -- John had correctly identified from the beginning. I believe they are in the first memo -- it was necessary, to make things work, to have some way of attaching the typewriters; some way of getting essentially a timer clock into the processor so that one could start the processor off on some job, and still have a way of recapturing the processor before the job finished. In those days the way most computers were organized the job would run to completion and then would return to the supervisor program. So you wanted, essentially, an egg-timer that would grab it back, and one you could set to some fairly tight timing mark -- a tenth of a second -- so that you could run a job for a quick burst, because that was important. If you were going to multiplex between a lot of different jobs, you needed to jump around a lot.

Another requirement was that we have some notion of memory bounds registers. Computers were designed, in those days, on the assumption that there would be only one program and that all parts of that program would be equally friendly and well-behaved. There wasn't this "we versus they" attitude that you really ought to have when you build an operating system, particularly the supervisor programs, which had evolved in software, but the hardware people did not recognize a need for that distinction yet. So they would allow user programs to do everything that a supervising program could do. In particular, the user programs could make two mistakes. One, they could trample on somebody else's program that might be in memory at the same time and get out of their bounds, either reading or writing, both sinful -- reading for privacy; writing for destruction. Secondly, they could issue privileged instructions, such as input-output to any device on the machine, which could create chaos. In a lot of the early design of timesharing systems, the easy way out was to say that the only person to issue input-output instructions should be the supervisor. So we had RPQs to try to withdraw I/O control, to have a mode of the machine where the users could not issue I/O instructions, and secondly, could not go beyond its bound registers. Also, the word length was such that there were a lot of unused instructions in the instruction field, the IBM instruction format, and they had left undefined what would happen on all those instructions. A lot of the people had come to start using gimmicky features, which were really not features. You could not predict what would happen. It was an incomplete design, in retrospect a bad design, again, based on the fallacy that people would write perfect programs (laugh). We wanted those also to trap to the instruct supervisor, so a person could not create unpredictable...

NORBERG: Was it possible back then that the companies, when building such a machine -- say, designing the 704 -- were working under the assumption that they would provide all the software anyway, and that therefore they could control the perfect characteristics of the software and not worry about this problem of poor design that you just mentioned?

CORBATO: That may have been true at some level of the company. I think most of the practicing programmers and field managers and the like already recognized already that they could not control programming. That was an

idea that died very fast. It had heen an idea that was true in the mid-1950s, but it was amazing how fast it went. In part that was their own doing. By introducing FORTRAN they convinced a lot of people that you did not have to be a professional programmer to write programs. But what they did think was that they knew how to run machines and that they knew hest the style of use. One of the reasons that we at MIT were so vociferous about that it could be otherwise was that many of us had cut our teeth on WHIRLWIND. WHIRLWIND was a machine that was like a hig personal computer, in some ways, although there was a certain amount of efficiency hatching and things. We had displays on them. We had typewriters, and one kind of knew what it meant to interact with a computer, and one still remembered. We could see we were losing that under the guise of efficiency of the machine. But what was obvious was that the efficiency of the people had gone down the tubes. The resistance we ran into was kind of a "Don't tell us how to view our husiness or how to huild machines." In particular, the architects of the equipment were more and more insulated from the use of their machines. They were infatuated with making the machines faster and did not recognize that there was a need for a modal change in the way it was dealt with.

NORBERG: Yes. Were there other centers where this sort of reaction might have taken place as well, people who had access to a machine that they were programming for themselves and therefore, might have felt the same as you people did? I am thinking of the federal labs, for example, hefore they became federal, of course. I am thinking of Los Alamos and its early computing facilities.

CORBATO: No, they were too well off. People's interest in the prohlem of timesharing was almost directly correlated with how short of computing equipment they were.

NORBERG: Yes. I remember you saying that before when the Annals meeting took place.